⑲日本国特許庁（JP）　⑪特許出願公開

⑫公開特許公報（A）　昭63－186338

⑤Int.Cl.⁴	識別記号	庁内整理番号	㊸公開　昭和63年(1988)8月1日
G 06 F 11/10	330	G－7368－5B	
H 03 M 13/00		6832－5J	

審査請求　未請求　発明の数 3　（全10頁）

㊹発明の名称　誤り訂正回路

㉑特　願　昭62－19301

㉒出　願　昭62(1987)1月28日

㉗発明者　冨満　康治　東京都港区芝5丁目33番1号　日本電気株式会社内

㉑出願人　日本電気株式会社　東京都港区芝5丁目33番1号

㉔代理人　弁理士　桑井　清一

明細書

1．発明の名称

誤り訂正回路

2．特許請求の範囲

（1）各々が並列ガロア体乗算回路と並列ガロア体加算回路と複数のレジスタとを含む複数のガロア体演算ユニットをバスにより直列に接続し、リード・ソロモン符号を含むＢＣＨ符号の生成と複合とを行うことを特徴とする誤り訂正回路。

（2）上記ガロア体演算ユニットは検査シンボル数と同数である特許請求の範囲第1項記載の誤り訂正回路。

（3）各々が並列ガロア体乗算回路と並列ガロア体加算回路と複数のレジスタとを含む複数のガロア体演算ユニットをバスにより直列に接続し、上記複数のガロア体演算ユニットを接続するバスとは別個に上記複数のガロア体演算ユニットに共通に接続されたバスをさらに有することを特徴とする誤り訂正回路。

（4）各々が並列ガロア体乗算回路と並列ガロア体加算回路と複数のレジスタとを含む複数のガロア体演算ユニットをバスにより直列に接続し、上記複数のガロア体演算ユニットを接続するバスとは別個に上記複数のガロア体演算ユニットに共通に接続されたバスを有し、上記複数のガロア体演算ユニットとは別個に並列ガロア体乗算回路と算術論理回路とレジスタとメモリとマイクロプログラム制御ユニットとを有し上記複数のガロア体演算ユニットに共通に接続されたバスに接続され上記複数のガロア体演算ユニットを制御するコントローラをさらに有することを特徴とする誤り訂正回路。

3．発明の詳細な説明

［産業上の利用分野］

本発明は誤り訂正回路に係り、特に多シンボルの誤りを訂正する誤り訂正回路に関する。

［従来の技術］

従来、この種の誤り訂正回路として種々の構成が提案されており、その内の３つの構成をまず説明する。

第１の構成例は汎用のマイクロプロセッサを含むマイクロプロセッサシステムで実現する例であり、誤り訂正の過程におけるガロア体の演算等を所定のプログラムに基づき実行するものである。

これに対して、第８図に示されている構成例は１組のガロア体乗算回路とガロア体加算回路とを含む専用のハードウエアの例であり、第８図中、２０は乗算回路、２２はメモリ、２３はマイクロプログラム制御ユニット、３０は単位遅延素子、３５は加算回路、３８はシンドロームジェネレータをそれぞれ示している。

更に、第９図に示されている構成例はシストリックアレイにより実現された例であり、第９図中、３８はシンドロームジェネレータを、３９は誤り位置多項式計算回路を、４０は誤り位置計算回路を、４１は誤りパターン計算回路を、４２は誤り修正回路をそれぞれ示している。

［発明が解決しようとする問題点］

上記従来の各構成にあっては以下に記す問題点をそれぞれ含んでいる。

すなわち、誤り訂正をマイクロプロセッサシステムで実現した場合には、誤り訂正の過程でガロア体の乗算等の演算を実行しなければならず、かかるガロア体の乗算等を汎用のマイクロプロセッサで実行しようとすると長時間を必要とし、一般に専用のハードウエアに比べ約１０倍の時間を要するという問題点があった。

一方、第８図に示した１組のガロア体乗算回路とガロア体加算回路とを含む専用ハードウエアによる構成は上記マイクロプロセッサシステムを使用した構成に比べて処理速度は向上するものの、それでもディジタル・ビデオ伝送系等の高速伝送には処理速度が低く、上記例示した高速伝送系には使用できないという問題点がある。

これに対して、第９図に示したシストリックアレイにより実現した例は処理速度が十分に高く、高速伝送系にも適用できるものの、回路規模が大きくなり、実用化、特に集積回路化が難しいという問題点があった。

本発明は上記各従来例の問題点に鑑み、処理速度が高く、しかも回路規模の小さな誤り訂正回路を提供することを目的にしている。

［問題点を解決するための手段］

本願第１発明に係る誤り訂正回路は各々が並列ガロア体乗算回路と並列ガロア体加算回路と複数のレジスタとを含む複数のガロア体演算ユニットをバスにより直列に接続して構成し、リード・ソロモン符号を含むＢＣＨ符号の生成と複合とを行うことを要旨としている。

上記第１発明に牽連する第２発明の誤り訂正回路は、各々が並列ガロア体乗算回路と並列ガロア体加算回路と複数のレジスタとを含む複数のガロア体演算ユニットをバスにより直列に接続し、さらに上記複数のガロア体演算ユニットを接続するバスとは別個に上記複数のガロア体演算ユニットに共通に接続されたバスをさらに有することを要旨としている。

更に、上記第１発明に牽連する第３発明の誤り訂正回路は、各々が並列ガロア体乗算回路と並列ガロア体加算回路と複数のレジスタとを含む複数のガロア体演算ユニットをバスにより直列に接続し、上記複数のガロア体演算ユニットを接続するバスとは別個に上記複数のガロア体演算ユニットに共通に接続されたバスを有し、上記複数のガロア体演算ユニットとは別個に並列ガロア体乗算回路と算術論理回路とレジスタとメモリとマイクロプログラム制御ユニットとを有し上記複数のガロア体演算ユニットに共通に接続されたバスに接続され上記複数のガロア体演算ユニットを制御するコントローラをさらに有することを要旨とする。

上記構成の本願第１、第２および第３発明は誤り訂正の各ステップにおいて共通のハードウエアを使用して誤りの訂正を実行することができる。

［実施例］

次に、本発明の実施例を図面に基づき説明する。

第１図は本発明の一実施例の全体構成を示すブロック図であり、１はユニットを示しており、複数のユニット１は直列接続バス２により直列に接続されている。また、全てのユニット１は共通データバス３に接続されており、この共通データバス３にはコントローラ５も接続されている。コントローラ５はコントロールバス４と「０」検出信号線６とを介しても各ユニット１に接続されている。

上記各ユニット１は第２図に詳示されているように乗算回路７と、加算回路８と、マルチプレクサ９、１０、１５、１６と、トライステートゲート１１、１７、１８と、制御レジスタ１２と、レジスタファイル１３と、「０」検出回路１４とを有しており、上記直列接続バス２はマルチプレクサ９の入力に、共通データバス３はマルチプレクサ１５の入力に、「０」信号検出線６は「０」検出回路１４の出力にそれぞれ接続されている。

これに対して、上記コントローラ５は第３図に詳示されているように乗算回路２０と、マルチプレクサ２１、２４と、メモリ２２と、マイクロプログラム制御ユニット２３と、算術論理回路（ＡＬＵ）２５と、レジスタファイル２６と、カウンタ２７と、コントローラ・データバス２８と、データバッファ２９とを備えており、マイクロプログラム制御ユニット２３は上記コントロールバス４及び「０」検出信号線６に接続されており、データバッファ２９は共通データバス３に接続されている。

次に、誤り訂正の方式について説明する。一般に、多シンボルの訂正をおこなうＢＣＨ符号（リード・ソロモン符号）の復号に関しては各種の方法が提案されている。例えば、ピータソンの方法、バーレンカップ・マッシーの方法、古賀の方法、ユークリッドの互除法等である。本実施例ではユークリッドの互除法を採用しており、このユークリッドの互除法により効率よく復号し、誤りを訂正するための方式は以下のステップ（１）乃至（５）を順次実行する。

すなわち、

ステップ（１）：シンドロームの演算、

ステップ（２）：誤り位置多項式の導出（誤り数の判断）、

ステップ（３）：誤り位置の導出、

ステップ（４）：誤りパターンの演算、

ステップ（５）：誤り訂正。

以下上記各ステップを順を追って説明する。

ステップ（１）：シンドロームの演算

一般にｔ重誤り訂正ＢＨＣ符号の検査行列Ｈは（１）式で与えられる。

（以下、余白）

$$H=\begin{bmatrix} 1 & 1 & \\ \alpha^{n-1} & \alpha^{n-2} & \\ \cdot & \cdot & \\ \cdot & \cdot & \\ \alpha^{(2t-2)(n-1)} & \alpha^{(2t-2)(n-2)} & \\ \alpha^{(2t-1)(n-1)} & \alpha^{(2t-1)(n-2)} & \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ \alpha^{2} & \alpha & 1 \\ \cdot & \cdot & \\ \cdot & \cdot & \\ \alpha^{2(2t-2)} & \alpha^{2t-2} & 1 \\ \alpha^{2(2t-1)} & \alpha^{2t-1} & 1 \end{bmatrix}$$

・・・・・・・（１）式

すなわち、符号語ａに対してａ＝（a_0, a_1, ・・・, a_{n-1}）とし、

$aH^T=0$ ・・・・・・・（２）式

符号語ａに誤りｅが加わり、受信語ｙ＝（ａ＋ｅ）

が復号器に入力され、シンドロームS＝（S_0, S_1, ・・・, S_{2t-1}）は（3）式に従い求められる。

$$S = y H^T$$

故に、

$$S_j = \sum_{i=0}^{n-1} \alpha^{j(n-1-i)} \cdot y_i \quad (j = 0 \sim 2t)$$

・・・・・（3式）

（$y = (y_0, y_1, \cdots, y_{2t-1})$

ステップ2：誤り位置多項式の導出

誤り位置多項式は（4）式により定義される。

$$\sigma(Z) = \prod_{i \in E} (Z - \alpha^i) = \sum_{j=0} \sigma_j Z^j \quad （4）式$$

ただしEを誤り位置の集合とする。

（4）式からも分かるように、誤り位置多項式を求めると、次数により誤りの数がわかり、根により誤りの位置がわかる。

このσ（Z）を求める方法としてユークリッドの互除法がある。

一般に次式が成立する。

$$\sigma(Z) S(Z) + \phi(Z) Z^{2t} = \eta(Z)$$

・・・・（5）式

ただし、η（Z）は誤り数値多項式で、（6）式に示される。

$$\eta(Z) = \sum_{i \in E} l_i \prod_{\substack{j \in E \\ j \neq i}} (Z - \alpha^j)$$

・・・・（6）式

また、S（Z）はシンドローム多項式であり、次式で表される。

$$S(Z) = -\sum_{j=0}^{2t-1} S_j \cdot Z^j$$

また、φ（Z）は

$$\phi(Z) = \sum_{i \in E} l_i \alpha_i^{-2t} \prod (Z - \alpha^j)$$

（5）式にこのままでS（Z）を与えても解は求まらない。しかしながら、誤り個数eがt以下のときには各誤りパターンは1つの異なるシンドローム多項式を持つから、S（Z）を与えてη（Z）とσ（Z）とを一意に求めることができる。

第4図にユークリッドの互除アルゴリズムにより（5）式の解を求めるフローチャート図を示す。

第4図に於て、［　］は除算の商を表しており、δは$U_i(z)$の最大次数の係数を表している。

ステップ3：誤り位置の検出

ユークリッドの互除アルゴリズムにより導かれたσ（Z）の根が誤り位置を与える。この根を求める方法としてチエンのアルゴリズムがある。この方法はσ（Z）にα^0からα^{n-1}を順に代入し、σ（α^j）＝0となる位置jが誤り位置になることを利用する。

ステップ4：誤りパターンの演算

ユークリッドの互除アルゴリズムにより導かれた誤り位置多項式σ（Z）、誤り数値多項式η（Z）およびチエンのアルゴリズムから得られるiにより誤り位置iでの誤り数値l_iを求める。

（4）式を形式微分すると

$$\sigma'(Z) = \sum_{i \in E} \prod_{\substack{j \in E \\ j \neq i}} (Z - \alpha^j) \quad ・・・（7）式$$

（7）式に誤り位置α^iを代入すると、

$$\sigma'(\alpha^i) = \prod_{\substack{j \in E \\ j \neq i}} (\alpha^i - \alpha^j) \quad ・・・（8）式$$

一方、（6）式にα^iを代入すると、

$$\eta(\alpha^i) = l_i \prod_{\substack{j \in E \\ j \neq i}} (\alpha^i - \alpha^j) \quad ・・・（9）式$$

（8）式と（9）式とに基づき、

$l_i = \eta(\alpha^i) / \sigma'(\alpha^i)$　・・（10）式

となり、誤りパターンが求められる。

ステップ5：誤り訂正

$y = a + e$　より　$a = y - e$となり、誤りの訂正がなされる。

次に、上記誤り訂正方式にしたがった一実施例の作用を説明する。今、メモリ22に受信語が入力されているとする。各ユニット1のレジスタファイル13の1つのレジスタにα^0, ・・・・・, α^{n-1}をそれぞれ供給し、次に、シンドロームの保持される別のレジスタに「0」を入力し、順次メモリ22から受信語をコントローラ内部バス28、データバッファ29を介して共通データバス3に

出力する。

一方、各ユニットの内部ではマルチプレクサ９が共通データバス３側に、マルチプレクサ１５が加算器８の出力側に、マルチプレクサ１０がレジスタの出力側にそれぞれ設定され、共通バスからのデータＸｉに対して、

$$Y_i = Y_{i-1}\alpha^j + X_i$$

の演算を繰り返す。

ただし、Ｙｉはｉ回目のレジスタの内容である。

ここで、Y_{n-1}は

$$Y_{n-1} = Y_{n-2}\alpha^j X_{n-1}$$
$$= (Y_{n-3}\alpha^j + X_{n-2})\alpha^j + X_{n-1} =$$
$$= \cdots$$
$$= \sum_{i=0}^{n-1} \alpha^{j(n-1-i)} X_i$$

となり、各ユニットのレジスタに（３）式で示したシンドロームが生成される。

次に、第４図で示したユークリッドのアルゴリズムにより誤り位置多項式（式（４））、誤り数値多項式（式（６））を求める。この演算の中心となるのが多項式同士の除算、乗算、加算である。除算回路を第５図に示す。第５図に於て３０は単位遅延素子、３１は加算回路、３２は係数乗算器をそれぞれ示している。第５図に示された回路は$b_0 + b_1 x + \cdots + b_n x^n$を被除数多項式とする除算回路であり、$a_0 + a_1 x + \cdots + a_n x^n$を除数多項式とした場合、$a_n$から順に入力すれば商の値は出力端子に現れ、剰余は最後にシフトレジスタに残される。

これに対して乗算回路は第６図に示されているように単位遅延素子３０と、加算回路３１と、係数乗算器３２とを有しており、第６図に示された回路は$b_0 + b_1 x + \cdots + b_n x^n$を被乗数多項式とし、$a_0 + a_1 x + \cdots + a_n x^n$を乗数多項式とした場合、$a_n$から順に入力すれば積の値は出力端子に現れる。それで、１組の加算器と遅延素子と係数器とを１ユニットに割り当てれば上記計算を実行することができる。

マルチプレクサ１０を介して入力された共通データバス上のデータとレジスタ１２内の係数とを乗算し、直列接続バスからマルチプレクサ９を介して入力された前段シフトレジスタの内容を加算し、マルチプレクサ１５を介してレジスタに入力される。このレジスタが第５図、第６図における遅延素子に相当する。出力されたデータは共通データバスを介していずれかのユニットのレジスタに格納される。除算の場合、出力の逆数の計算はコントローラ５内のメモリ２２の逆数テーブルによりなされ、再度、その逆数を共通データバスに出力することによりなされる。一方、乗算の場合、入力するデータは該当するレジスタから共通データバス３に出力され、コントローラ５内のデータバッファ２９にラッチされた後に再度、共通バスに出力されて各ユニットに入力される。除算の場合には、出力データの逆数を計算する際に、該当するレジスタから共通データバス３に出力された後に初めのユニットにフィードバックされる。

各ユニットのレジスタ出力には「０」検出回路１３が設けられており、この出力はマイクロプログラム制御ユニット２３に入力され、コントローラ５が各多項式の次数を知ることができるようになっている。すなわち、第４図における、

$$\deg R(Z) \leq t-1$$

の判断もこの方法による。加算、および

$$\sigma(Z) \leftarrow U_1(Z) / \delta$$
$$\eta(Z) \leftarrow R(Z) / \delta$$

は各ユニット内の加算または乗算（逆数を得た後）により実行する。以上の方法により$\sigma(Z)$および$\eta(Z)$を求めることができる。

次に、チエンのアルゴリズムの実行について説明する。チエンのアルゴリズムは第７図に示された回路により実行される。第７図に示された回路は単位遅延素子３３と、係数乗算器３４と、加算回路３５と、「０」検出回路３６と、「０」検出出力ノード３７とを有しており、$\sigma(Z)$の係数を初期値として各係数に順次α^0、α^1、・・・、α^{n-1}を掛け、全ての積を加算してその結果が「０」であった場合、その繰り返し回数が誤り位置となる。

ユニット1でこれを実現するには2つのレジスタからσ（Z）の係数とα^{0}、α^{1}、・・・・・・、α^{n-1}を出力し、乗算を実行し、マルチプレクサ9を直列接続バスに制御し、この結果を順次加算回路に供給して行けば計算結果が得られる。ユニットの数はt重誤り訂正の場合、2t個になるが、σ（Z）はt次以下になるので、その最高次の次に接続されるユニットのレジスタにつながる「0」検出回路13により結果が「0」になったことを確認する。繰り返し回数はコントローラ5内のカウンタ27によりカウントされ、結果が「0」になるとこの内容はメモリ22に転送される。以上により、最終的に目盛り22に誤り位置が全て格納される。

次に、誤りパターンの計算を行う。（10）式の分子η（α^{i}）はシンドロームと同様に

$$Y_{i-1}=\eta i+\alpha^{i}\cdot Y i$$

（$\eta(Z)=\sum_{i=0}\eta i\cdot Z^{i}$とする）

繰り返し演算可能である。

ことができる。

以下、式（10）の演算はコントローラ5の内部で行われ、誤り訂正の実施もコントローラ5の内部で実行される。

［発明の効果］

以上説明してきたように、本発明では各々が加算器と乗算器とを含むユニットで並列処理を行うので、1組の加算器と乗算器としか持たないシステムより数倍から数十倍の高速処理が可能になった。

また、シストリックアレイのように各ステップ毎にハードウエアを用意する方式に比べると、本発明では各ステップで共通のハードウエアを使用するので、回路の規模を縮小させることができる。具体的には各ユニットを1000ゲートで実現できるので、CMOSを使用して1ゲートを4トランジスタで構成しても集積回路化が十分に可能である。　さらに、t重誤りの訂正に対しても2t個のユニットを準備すればよく、拡張性においても利点がある。

η（Z）は第1図に於て、左側のユニットのレジスタから低次の係数が格納されているので、トライステートゲート17を高インピーダンスにしておき、ここからデータを入力してマルチプレクサ10を介して乗算回路7に入力する。もう一方の入力はコントローラ5のメモリ22から転送された誤り位置が格納されたレジスタの内容を入力し、これを乗算し、レジスタファイル13の出力からマルチプレクサ9を介してηiを出力し、加算回路8から出力し、この結果をトライステート18から直列接続バスに出力される。以上のステップによりη（α^{i}）を求めることができる。

次に、分母の演算であるが、ある誤り位置に対して残りの誤り位置との和の積となるから、コントローラ5からまず順次誤り位置を出力順に各ユニットのあるレジスタに格納する。次に、再度コントローラ5から順次誤り位置を出力し、各ユニット内にある誤り位置と加算して同一ユニット内の別のレジスタに格納する。次に、各々の和を直列接続バスを介して乗算すれば上記分母を求める

4．図面の簡単な説明

第1図は一実施例の構成を示すブロック図、

第2図はユニットの構成を示すブロック図、

第3図はコントローラの構成を示すブロック図、

第4図はユークリッドの互除アルゴリズムを示すフローチャート図、

第5図はGF（2^{n}）の多項式の除算回路のブロック図、

第6図はGF（2^{n}）の多項式の乗算回路を示すブロック図、

第7図はチエンのアルゴリズムの実行回路を示すブロック図、

第8図は従来の誤り訂正回路を示すブロック図、

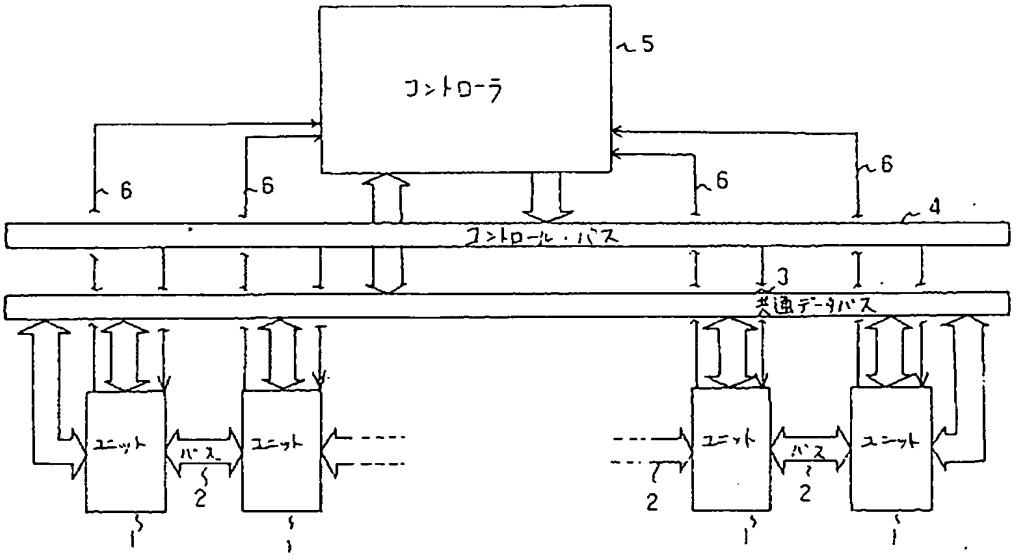
第9図は従来の他の誤り訂正回路を示すブロック図である。

1・・・・・・・ユニット、

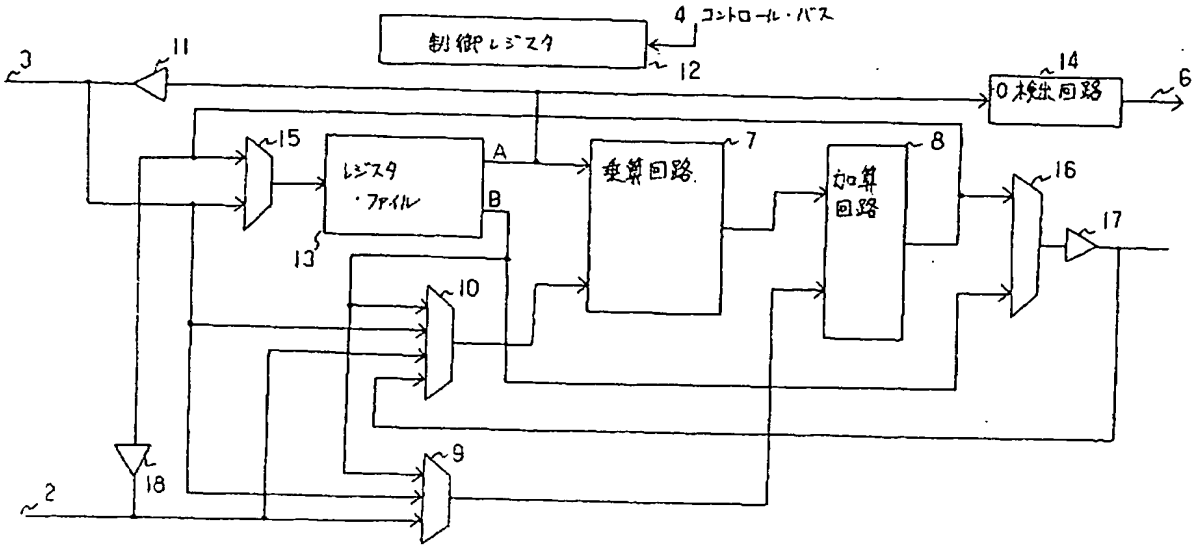
2・・・・・・・直列接続バス、

3・・・・・・共通データバス、
4・・・・・・コントロールバス、
5・・・・・・コントローラ、
6・・・・・・「0」検出出力、
7・・・・・・乗算回路、
8・・・・・・加算回路、
9、10・・・・マルチプレクサ、
11、17、18
・・・・トライステートゲート、
12・・・・・制御レジスタ、
13・・・・・レジスタファイル、
14・・・・・「0」検出回路、
15、16・・・マルチプレクサ、
19・・・・・レジスタ、
20・・・・・乗算回路、
21・・・・・マルチプレクサ、
22・・・・・メモリ、
23・・・・・マイクロプログラム制御
ユニット、
24・・・・・マルチプレクサ、
25・・・・・ALU、
26・・・・・レジスタファイル、
27・・・・・カウンタ、
28・・・・・コントローラデータバス、
29・・・・・データバッファ、
30、33・・・単位遅延素子、
31・・・・・加算回路、
32、34・・・係数乗算器、
35・・・・・加算回路、
36・・・・・「0」検出回路、
37・・・・・「0」検出出力、
38・・・・・シンドロームジェネレータ、
39・・・・・誤り位置多項式計算回路、
40・・・・・誤り位置計算回路、
（チエンのアルゴリズム
実行回路）、
41・・・・・誤りパターン計算回路、
42・・・・・誤り修正回路、
43・・・・・制御出力。

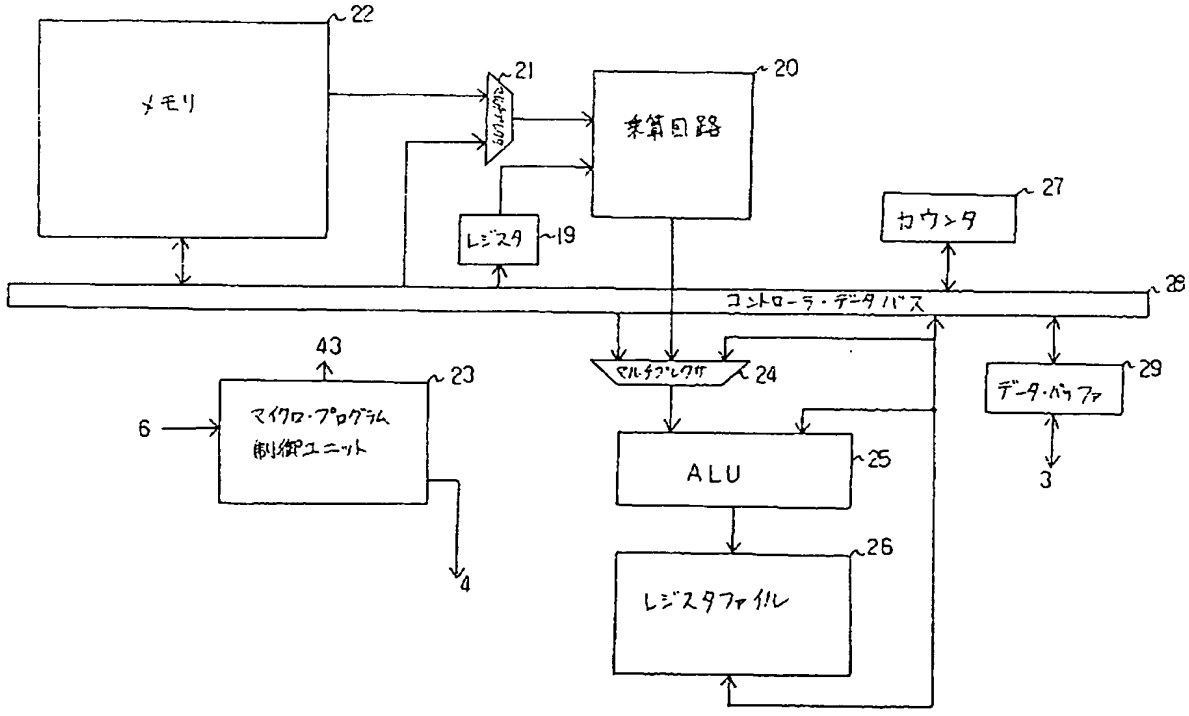
第1図



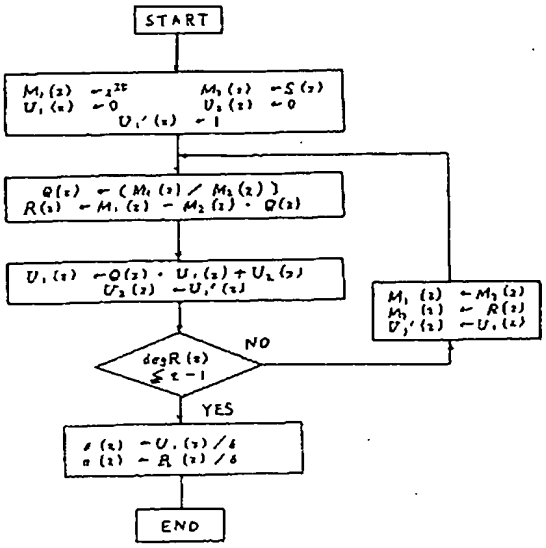
第2図


制御レジスタ
4 コントロール・バス
12
3
11
14
0検出回路
6
15
レジスタ・ファイル
A
B
13
乗算回路
7
加算回路
8
16
17
10
9
18
2

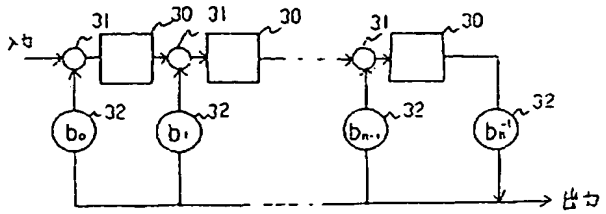
第3図


メモリ
22
21
マルチプレクサ
乗算回路
20
レジスタ
19
カウンタ
27
コントローラ・データバス
28
43
マイクロ・プログラム制御ユニット
23
6
マルチプレクサ
24
データ・バッファ
29
3
ALU
25
レジスタファイル
26
4

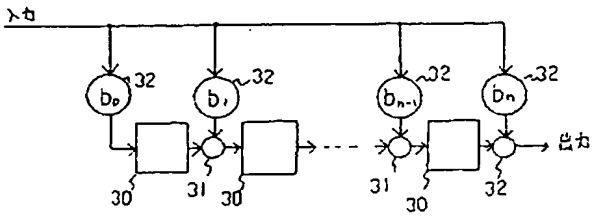
第4図



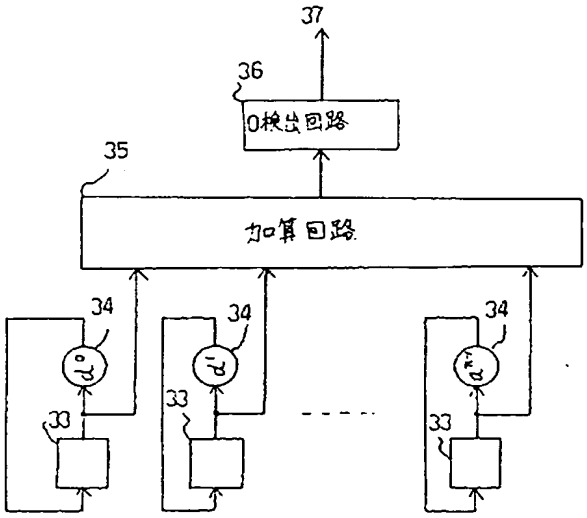
第5図



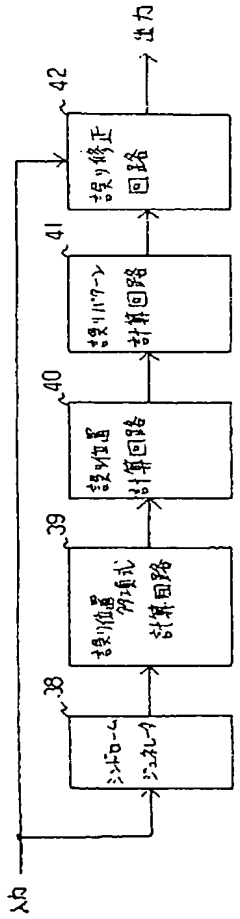
第6図



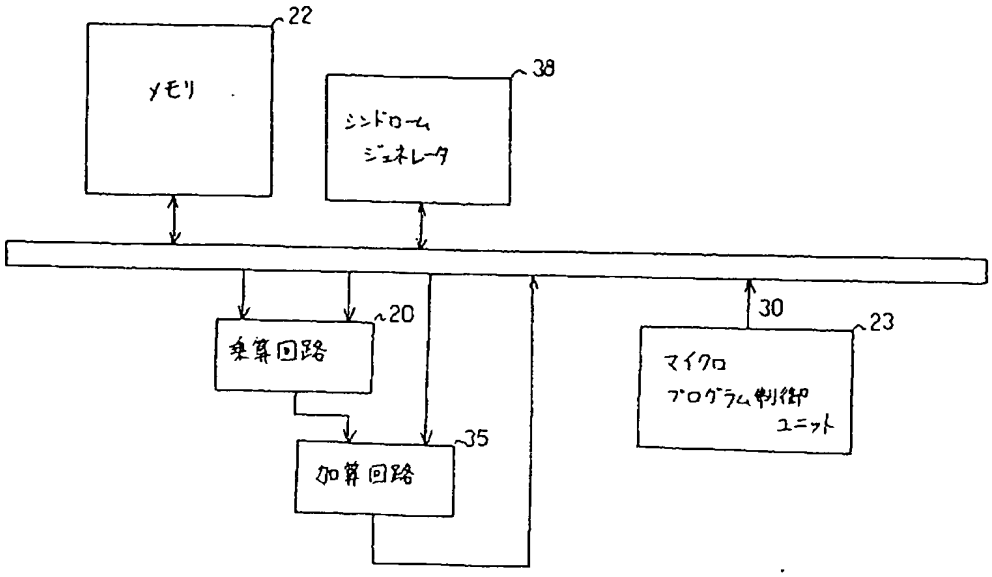
第7図



第9図



第8図


22
メモリ
38
シンドローム
ジェネレータ
20
乗算回路
35
加算回路
30
23
マイクロ
プログラム制御
ユニット